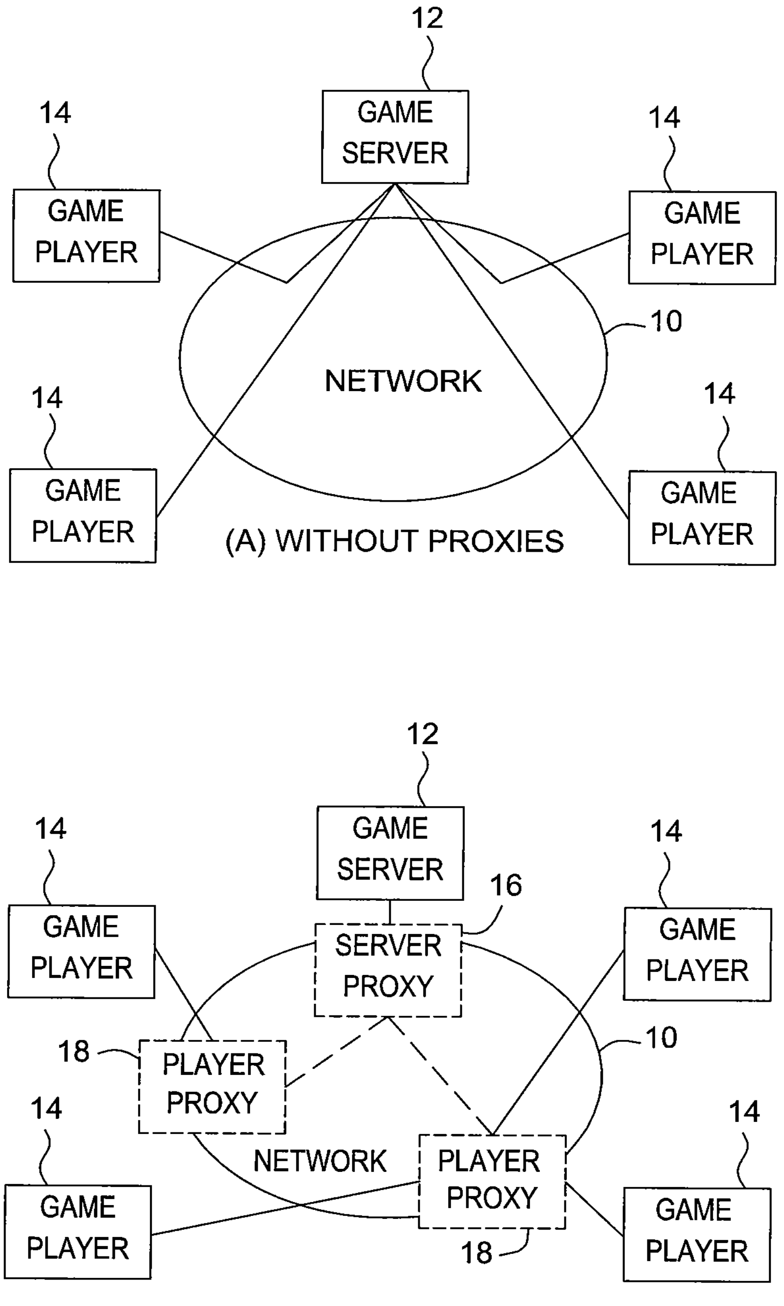


(A) WITHOUT PROXIES

(A) WITH SERVER AND PLAYER PROXIES

DISTRIBUTED GAME ENVIRONMENT

FIG. 1

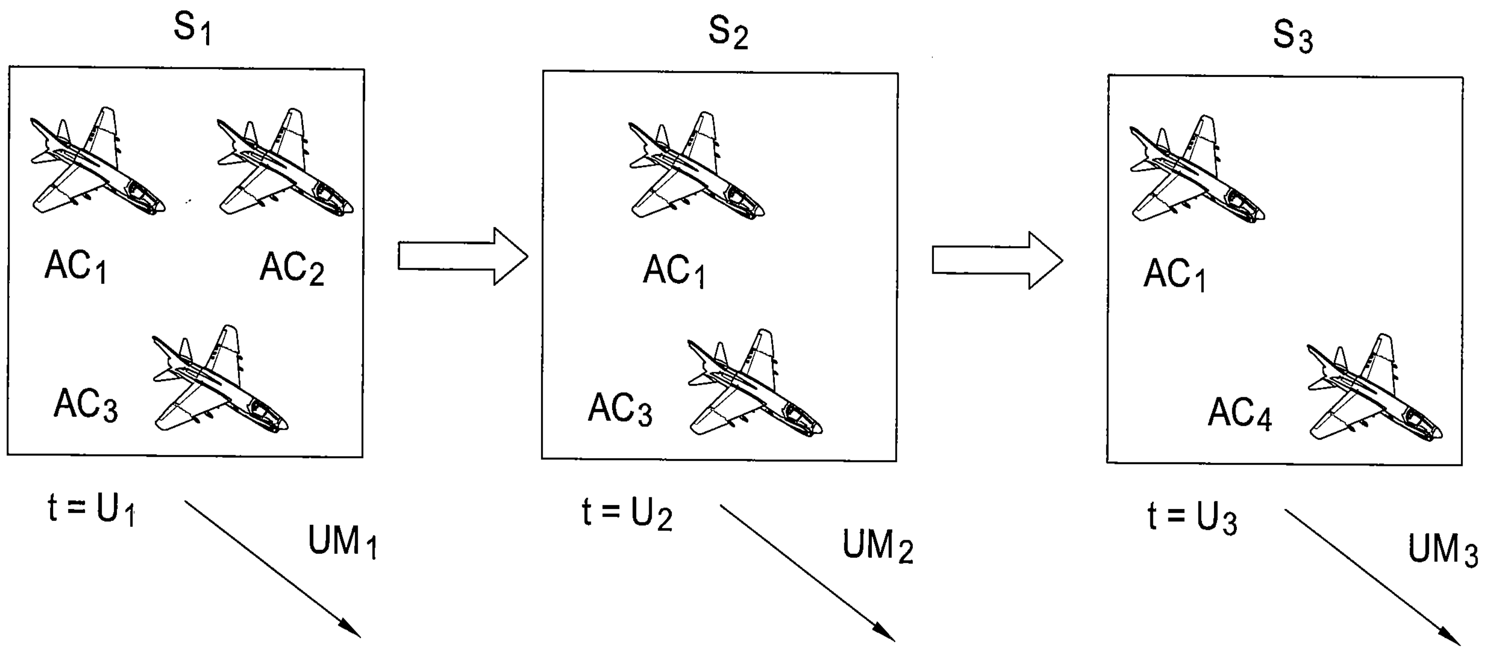
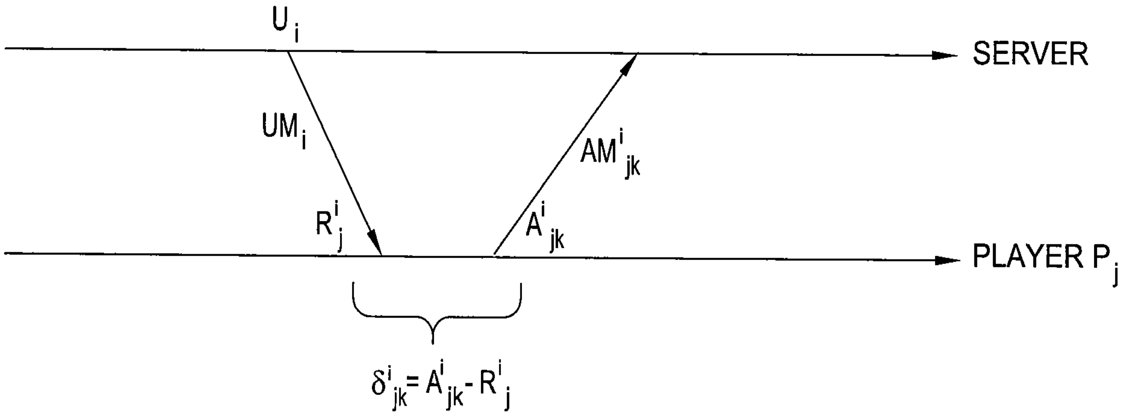


FIG. 2



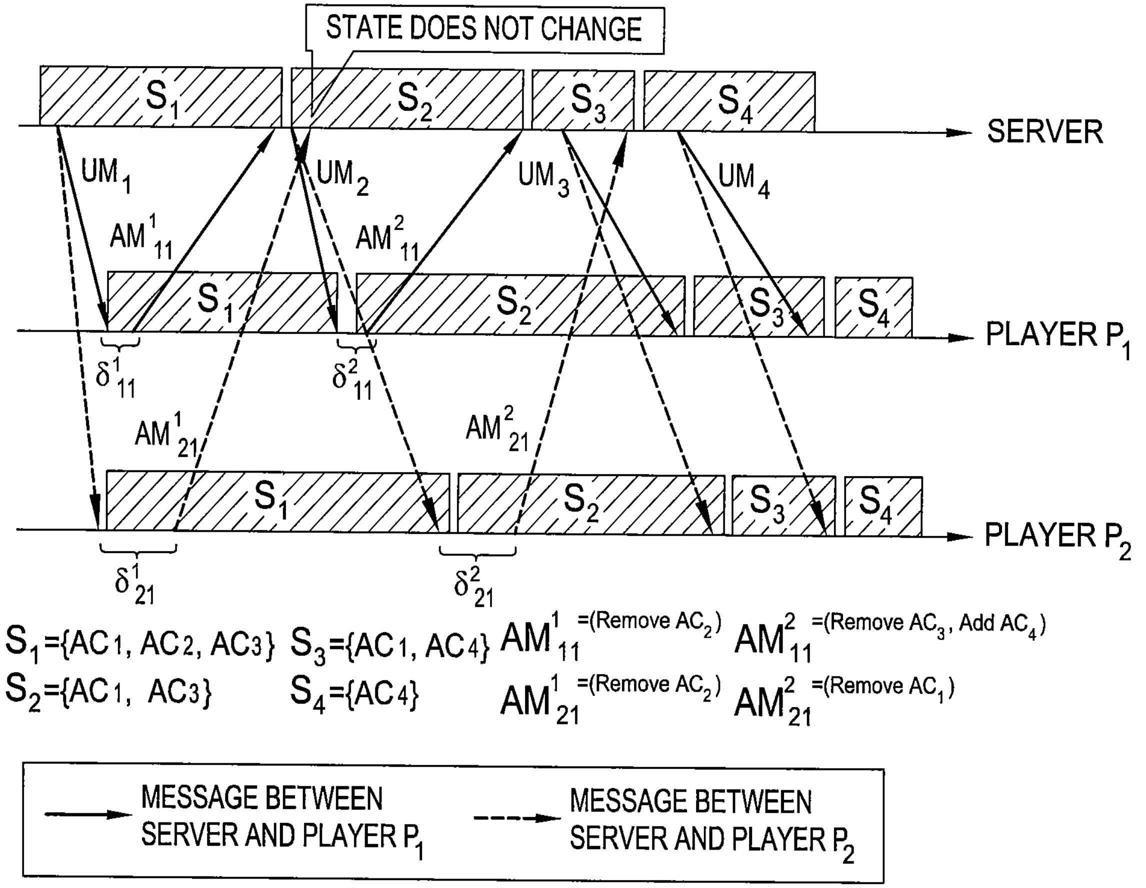
MESSAGE EXCHANGE BETWEEN SERVER AND PLAYERS

FIG. 3

```
ALGORITHM FAIR-ORDER MESSAGE QUEUEING (ACTION_MESSAGE $M_k$):
1: COMPUTE $D(M_k)$ = DELIVERY TIME OF $M_k$;
2: INSERT $M_k$ INTO DELIVERY QUEUE SORTED ACCORDING TO $D(M_k)$;
3: IF (DELIVERY QUEUE SIZE > 1)
4:        RECOMPUTE DELIVERY TIME OF EXISTING MESSAGES;
```

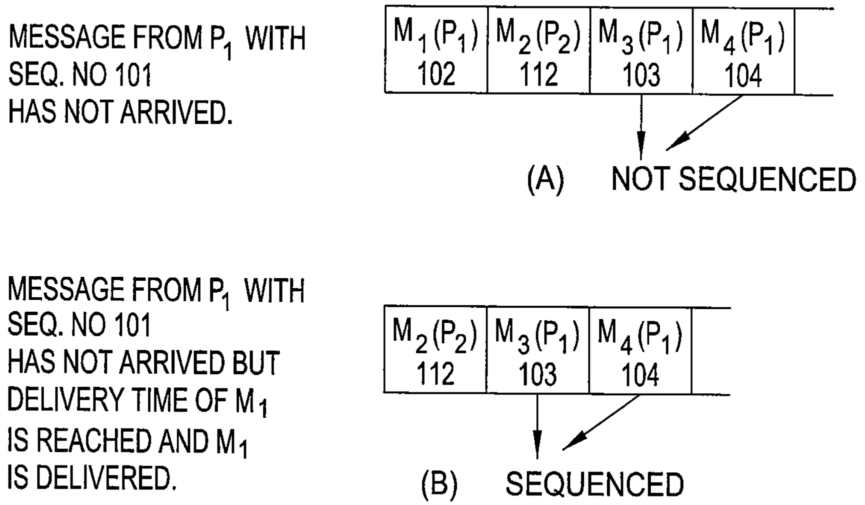
ALGORITHM FOR FAIR-ORDER MESSAGE QUEUEING

FIG. 5



FAIR-ORDER MESSAGE DELIVERY FOR
STATE TRANSITIONS SHOWN IN FIG. 2

FIG. 4



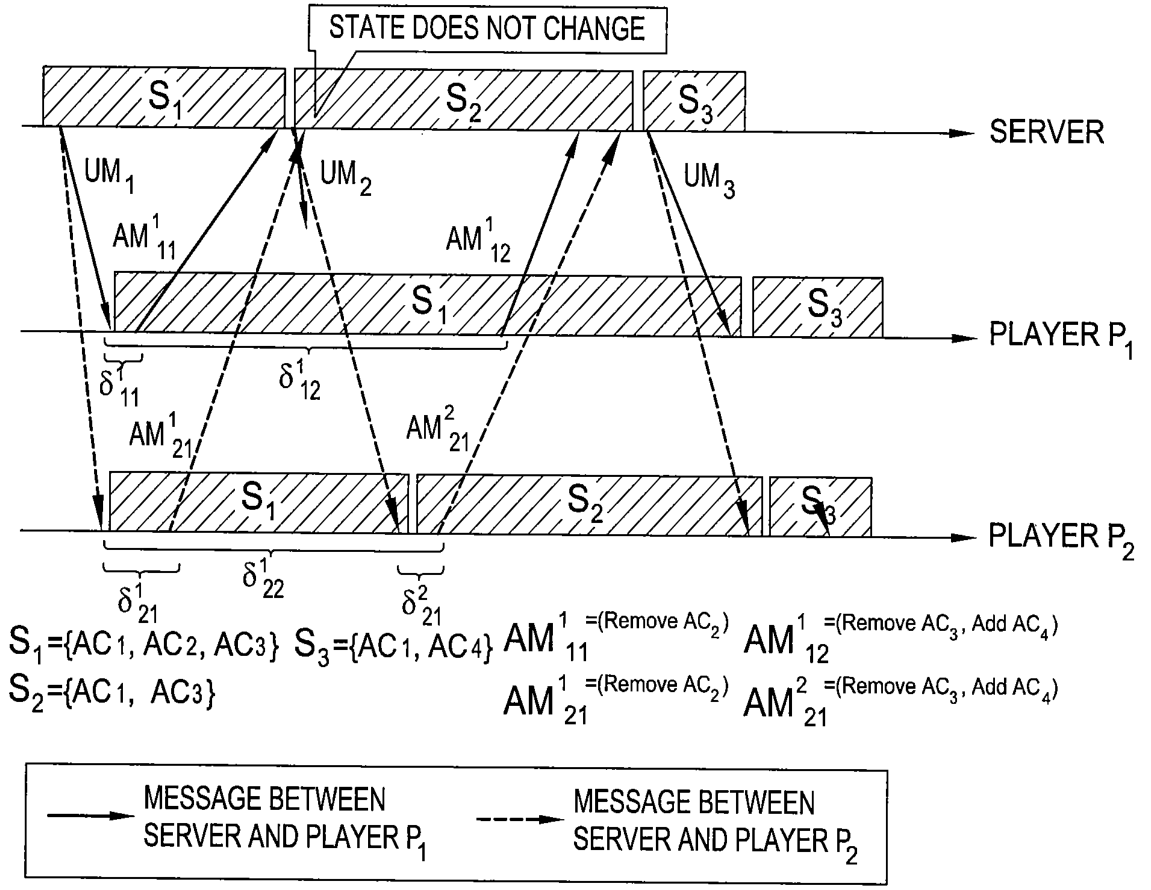
EXAMPLE WHERE MESSAGES ARRIVE AFTER THEIR WAIT TIMEOUT.

FIG. 6

```
ALGORITHM FAIR-ORDER MESSAGE DEQUEUING (ACTION_MESSAGE $M_k$):
1: DELIVERY $M_k$ AT $D(M_k)$;
2: IF (DELIVERY QUEUE SIZE > 1)
3:        RECOMPUTE DELIVERY TIME OF EXISTING MESSAGES;
```

ADDITIONAL ALGORITHM FOR FAIR-ORDER MESSAGE DEQUEUING WHEN MESSAGES DO NOT ARRIVE WITHIN THEIR WAIT TIMEOUT.

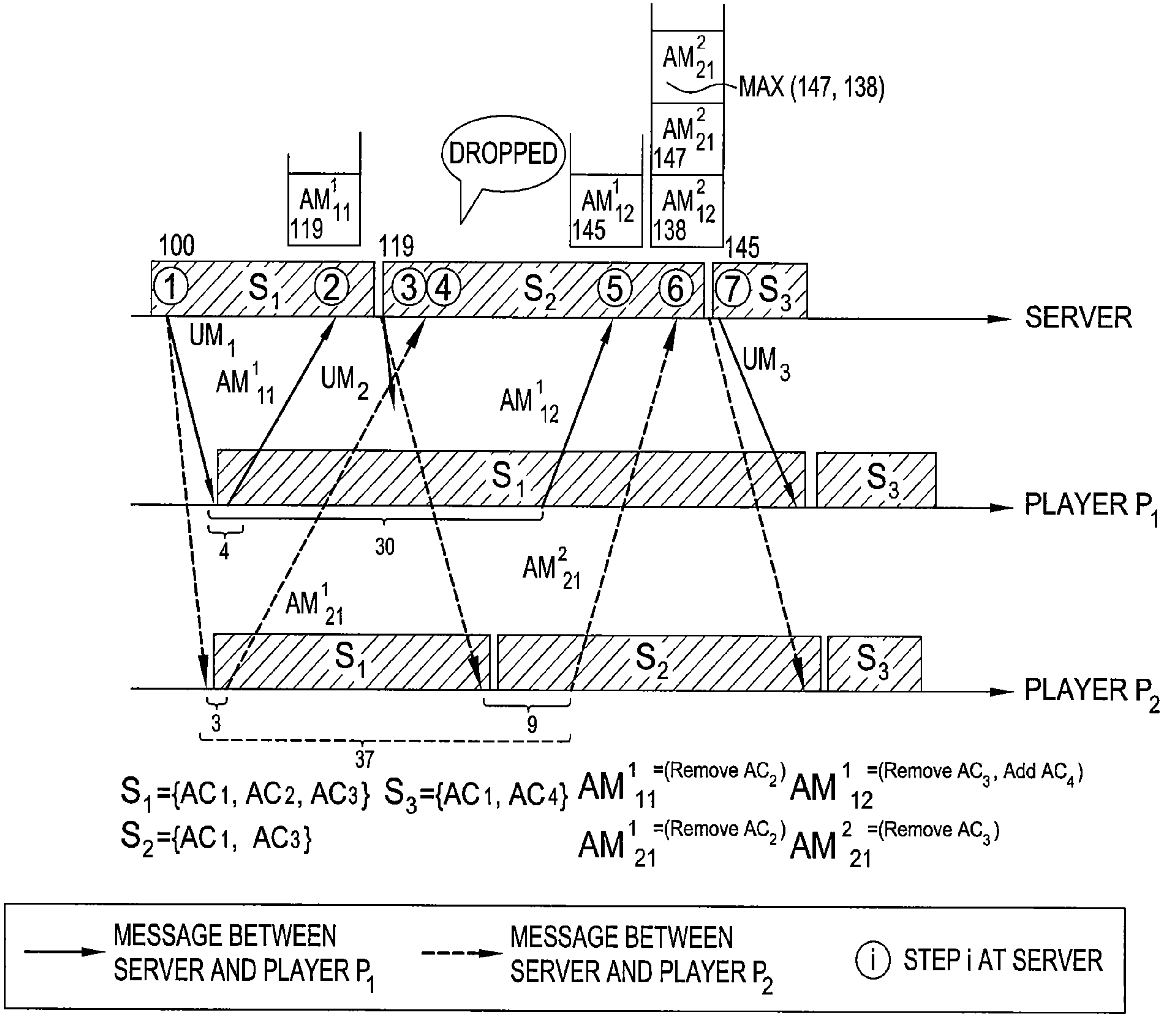
FIG. 7


STATE DOES NOT CHANGE
S1
S2
S3
SERVER
UM1
UM2
UM3
AM¹11
AM¹12
S1
S3
PLAYER P1
δ¹11
δ¹12
AM¹21
AM²21
S1
S2
S3
PLAYER P2
δ¹21
δ¹22
δ²21
S1={AC1, AC2, AC3} S3={AC1, AC4} AM¹11 =(Remove AC2) AM¹12 =(Remove AC3, Add AC4)
S2={AC1, AC3} AM¹21 =(Remove AC2) AM²21 =(Remove AC3, Add AC4)
MESSAGE BETWEEN SERVER AND PLAYER P1
MESSAGE BETWEEN SERVER AND PLAYER P2

EXAMPLE OF AN INCONSISTENT VIEW OF THE GAME
BETWEEN TWO PLAYERS, WHERE THE SEQUENCE OF STATE CHANGES
AT THE SERVER IS SHOWN IN FIG. 2

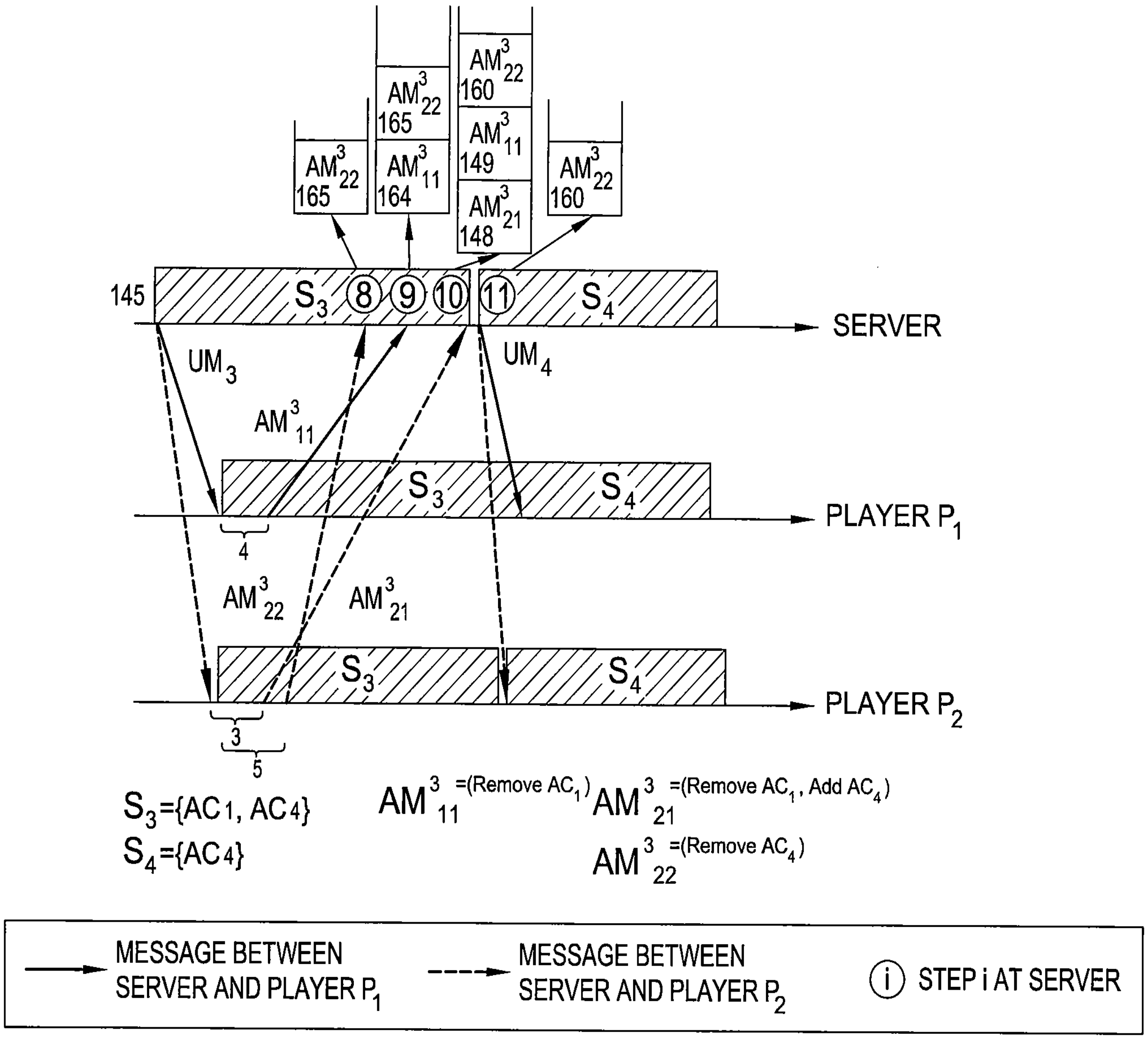
FIG. 8

FIG. 9



EXAMPLE SHOWING THE FAIR-ORDER MESSAGE DELIVERY ALGORITHM.

FIG. 10



EXAMPLE SHOWING THE FAIR-ORDER MESSAGE DELIVERY ALGORITHM WITH OUT-OF-ORDER MESSAGE RECEPTION.

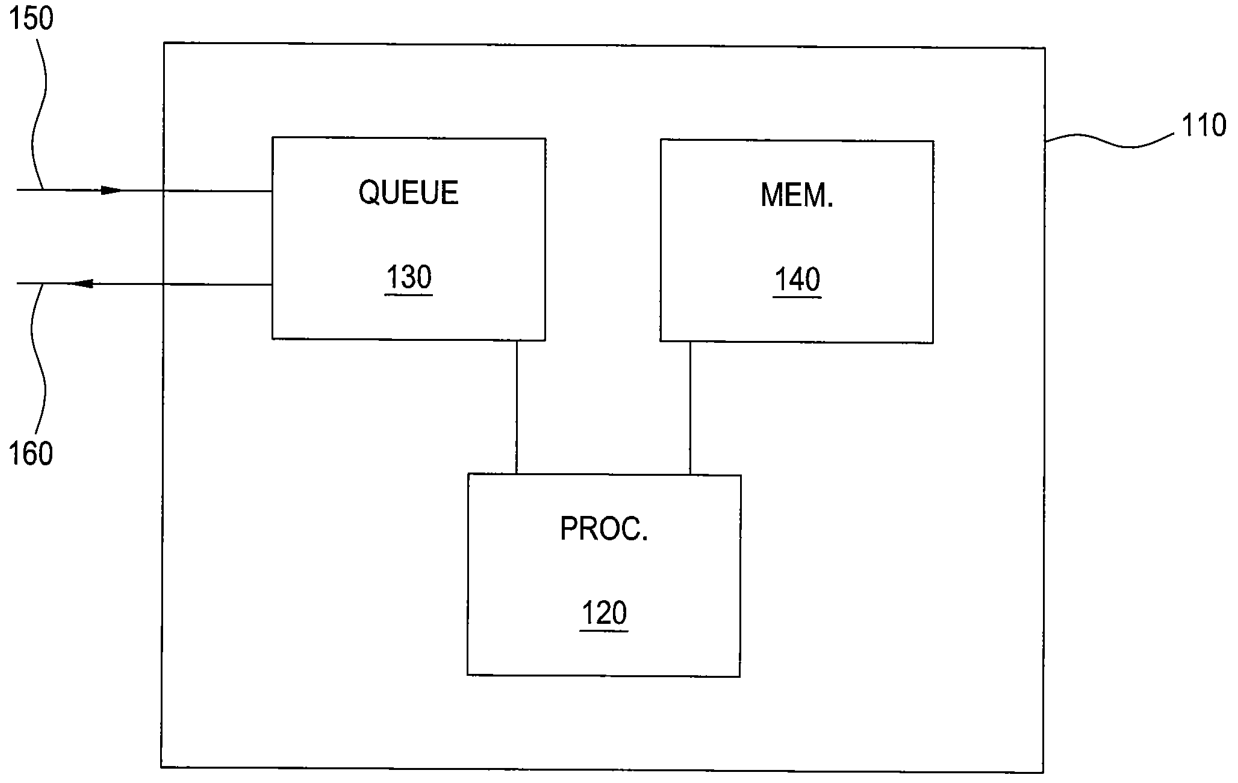


FIG. 11